CS 188 Introduction to AI Spring 2005 Stuart Russell — Final Solution

1. (12 pts.) Some Easy Questions to Start With

(a) (2) *False.* DBNs can include continuous variables.

(b) (2) *True.* The two clauses resolve to give $Q(F(F(z)))$, which entails $Q(F(F(G(A))))$ for any G, A.

(c) (2) *False.* α might just be consistent with β, e.g., $\alpha = (A \vee B)$ and $\beta = (A \vee \neg B)$.

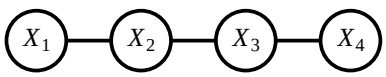
(d) (2) *True.* In this case, α either specifies a complete model (in which case β's truth value is fixed by it) or α contains contadictory unit clauses, in which case it is equivalent to $False$ and entails β anyway.

(e) (2) *True.* The states and actions are identical (except goals map to terminal states in the MDP), the transition model is deterministic, and the reward function is just minus the cost function.

(f) (2) *True.* It helps to overcome local ambiguity in the signal, just as with speech.

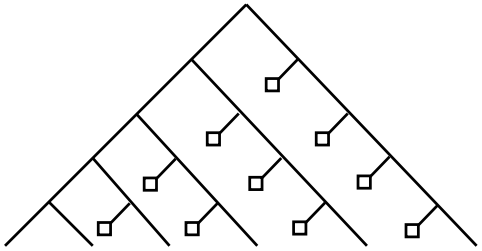
2. (10 pts.) Propositional Logic and CSPs

(a) (2)



(b) (2) (v) $n+1$ solutions. Once any X_i is true, all subsequent X_js must be true. Hence the solutions are i falses followed by $n-i$ trues, for $i = 0, \ldots, n$.

(c) (2) (iii) quadratic in n. This is somewhat tricky. Consider what part of the complete binary tree is explored by the search. The algorithm must follow all solutions sequences, which themselves cover a quadratic-sized portion of the tree. Failing branches are all those trying a $false$ after the preceding variable is assigned $true$. Such conflicts are detected immediately, so the tree looks like this (conflicts shown as squares):



(d) (1) *True.* Forward-chaining algorithm given in book.

(e) (1) *True.* Directed arc-conistency algorithm given in book.

(f) (2) *False.* Horn-form SAT problems do not necessarily map into tree-structured CSPs. (This was stated in lecture.)

3. (20 pts.) First-Order Logic following vocabulary: $IsA(x, c)$ means that object x is a member of product category c; $Weight(x, w)$ means that object x weighs w grams; $SKU1286$ is the name of a particular category of aluminium bolts.

(a) (2) $\forall x\ IsA(x, SKU1286) \Rightarrow Weight(x, 18)$.

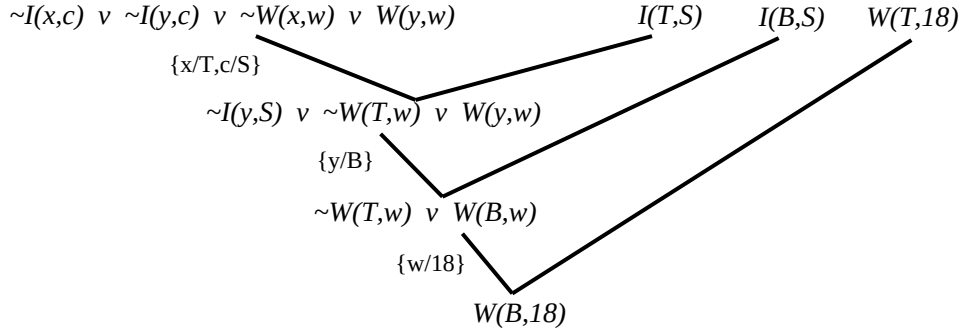
(b) (2) (ii) linear in m. The query unifies with the RHS of all m rules, so all m premises must be tested.

(c) (2) (i) constant time. The asserted fact unifies with only one rule premise.

(d) (2) (iii) linear in n. All n properties will be asserted for the object.

(e) (3) $IsA(x, c) \wedge IsA(y, c) \wedge Weight(x, w) \Rightarrow Weight(y, w)$.
It is not necessary to force x and y to be distinct: the case where $x = y$ is a tautology.

(f) (6) The proof looks like this. (Can do as refutation also.)



(g) (3) (i) constant time. Now the query matches just one rule, and each premise can be solved in constant time by lookup.

4. (20 pts.) Bayesian networks

(a) (3) (iii). The equation describes absolute independence of the three genes, which requires no links among them.

(b) (3) (i) and (ii). The *assertions* are the *absent* links, and (iii) asserts independence of genes which contradicts the inheritance scenario.

(c) (2) (i) is best. (ii) has spurious links among the H variables, which are not directly causally connected in the scenario described. (In reality, handedness may also be passed down by example/training.)

(d) (4) Notice that the $l \rightarrow r$ and $r \rightarrow l$ mutations cancel when the parents have different genes, so we still get 0.5.

G_{mother}	G_{father}	$P(G_{child}=l\|\ldots)$	$P(G_{child}=r\|\ldots)$
l	l	$1-m$	m
l	r	0.5	0.5
r	l	0.5	0.5
r	r	m	$1-m$

(e) (4) This is a straightforward application of conditioning:

$$\begin{aligned} P(G_{child}=l) &= \sum_{g_m,g_f} P(G_{child}=l|g_m,g_f)P(g_m,g_f) \\ &= \sum_{g_m,g_f} P(G_{child}=l|g_m,g_f)P(g_m)P(g_f) \\ &= (1-m)x^2 + 0.5x(1-x) + 0.5(1-x)x + m(1-x)^2 \\ &= x^2 - mx^2 + x - x^2 + m - 2mx + mx^2 \\ &= x + m - 2mx \end{aligned}$$

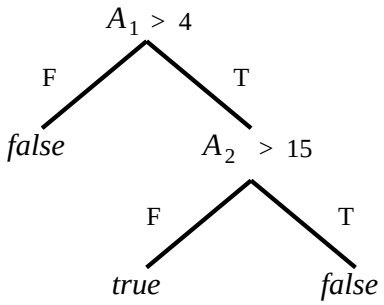
(f) (4) Equilibrium means that $P(G_{child}=l)$ (the prior, with no parent information) must equal $P(G_{mother}=l)$ and $P(G_{father}=l)$, i.e.,

$$x + m - 2mx = x, \text{ hence } x = 0.5.$$

But few humans are left-handed ($x \approx 0.08$ in fact), so something is wrong with the symmetric model of inheritance. The "high-school" explanation is that the "right-hand gene is dominant," i.e., preferentially inherited, but current studies suggest also that handedness is not the result of a single gene and may also involve cultural factors. See http://www.well.ox.ac.uk/~clyde.

5. (20 pts.) Learning

(a) (4) There are many possible trees. Here is one.



(b) (2) With 2 examples of each kind, the initial entropy is 1 bit. After the test, we have one subset with counts 0,1 (hence zero entropy) and one subset with counts 2,1. Hence the information gain is

$$1 - ((1/4) \cdot 0 + (3/4)(-1/3 \log(1/3) - 2/3 \log(2/3)) = 1 + (1/4) \log(1/3) + (1/2) \log(2/3) \approx 0.3113 bits.$$

(c) (4) Suppose the split gives two buckets with p_1, n_1 and $(p - p_1), (n - n_1)$ positive and negative examples respectively. Information gain is increased by moving these ratios towards 0 or 1. If the split is between two examples with the same classification, we can move it left or right to improve the left bucket by increasing or decreasing p_1 (if they are positive) or n_1 (if they are negative) as desired. This will also have the desired effect on the counts in the right bucket.

(d) (2) *False.* E.g., a test-once tree with one attribute creates exactly two regions on the real line, whereas the data may alternate between +ve and -ve more than once.

(e) (4) *Yes.* A test-many tree can define arbitrarily small hyper-rectangles, each containing exactly one example.

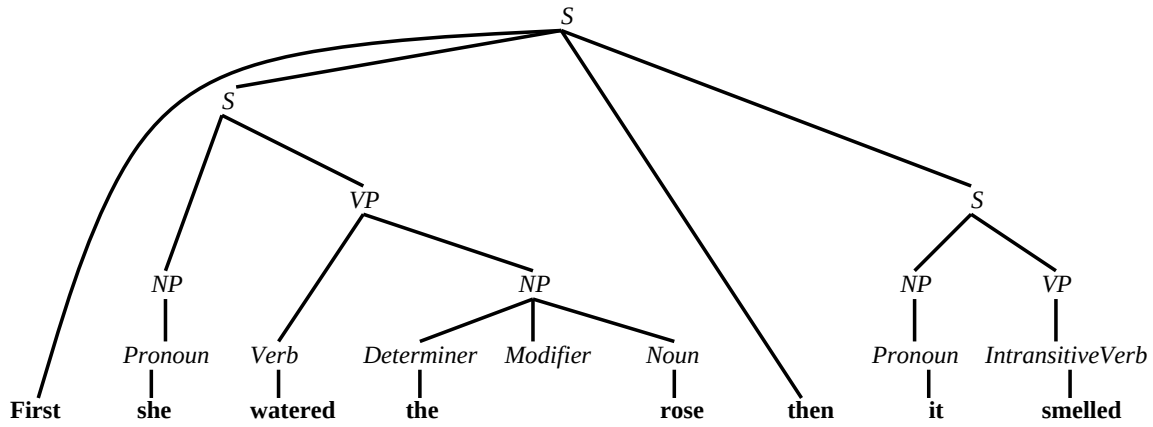
(f) (2) (i) (iii) (iv). Although (ii) can be classified, it requires many thin horizontal or vertical strips. The problem is that every split must be axis-parallel.

(g) (2) (i) (ii). The other two are not linearly separable.

6. (18 pts.) Natural language

(a) (6) (i) (iii). The grammar cannot generate (ii) because the modifier must be "red red rose", which is neither Adjective* nor Noun*.

(b) (4)



(c) (2) (iv) 4. Each "violet violet" modifier can be either Adjective* or Noun*, and 2 times 2 = 4.

(d) (1) (i) lexical. It arises from multiple meanings of "violet".

(e) (4) Again, several possibilities. The most obvious seems to be
S → first NP VP ThenClause$^+$
ThenClause → then NP VP

(f) (1) *False.* A word can have multiple meanings in the same syntactic category; a pronoun reference can be ambiguous; etc.